R E P U B L I Q U E F R A N Ç A I S E

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

-3 01108
(8)

BREVET D'INVENTION

CERTIFICAT D'UTILITÉ - CERTIFICAT D'ADDITION

COPIE OFFICIELLE

Le Directeur général de l'Institut national de la propriété industrielle certifie que le document ci-annexé est la copie certifiée conforme d'une demande de titre de propriété industrielle déposée à l'Institut.

Fait à Paris, le 13 JAN. 2004

Pour le Directeur général de l'Institut
national de la propriété industrielle
Le Chef du Département des brevets

Martine PLANCHE

INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE
SIEGE
26 bis, rue de Saint Petersbourg
75800 PARIS cedex 08
Téléphone : 33 (0)1 53 04 53 04
Télécopie : 33 (0)1 53 04 45 23
www.inpi.fr

INPI
INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE
26 bis, rue de Saint Pétersbourg
75800 Paris Cedex 08
Téléphone : 33 (1) 53 04 53 04 Télécopie : 33 (1) 42 94 86 54

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

Code de la propriété intellectuelle - Livre VI

cerfa N° 11354*03

REQUÊTE EN DÉLIVRANCE page 1/2

BR1

Cet imprimé est à remplir lisiblement à l'encre noire DB 540 e W / 210502

Réservé à l'INPI

REMISE DES PIÈCES
DATE 24 JAN 2003
LIEU 75 INPI PARIS F

N° D'ENREGISTREMENT NATIONAL ATTRIBUÉ PAR L'INPI 0301108

DATE DE DÉPÔT ATTRIBUÉE PAR L'INPI 24 JAN. 2003

1 NOM ET ADRESSE DU DEMANDEUR OU DU MANDATAIRE À QUI LA CORRESPONDANCE DOIT ÊTRE ADRESSÉE

JEUNE Pascale

FRANCE TELECOM R&D/VAT/PI
38-40, rue du Général Leclerc
92794 ISSY MOULINEAUX Cédex 9

Vos références pour ce dossier *(facultatif)* 04539

Confirmation d'un dépôt par télécopie ☐ N° attribué par l'INPI à la télécopie

2 NATURE DE LA DEMANDE — **Cochez l'une des 4 cases suivantes**

Demande de brevet	☒	
Demande de certificat d'utilité	☐	
Demande divisionnaire	☐	
Demande de brevet initiale	N°	Date
ou demande de certificat d'utilité initiale	N°	Date
Transformation d'une demande de brevet européen *Demande de brevet initiale*	☐ N°	Date

3 TITRE DE L'INVENTION (200 caractères ou espaces maximum)

Procédé cryptographique à clé publique pour la protection d'une puce contre la fraude

4 DÉCLARATION DE PRIORITÉ OU REQUÊTE DU BÉNÉFICE DE LA DATE DE DÉPÔT D'UNE DEMANDE ANTÉRIEURE FRANÇAISE

Pays ou organisation — Date — N°
Pays ou organisation — Date — N°
Pays ou organisation — Date — N°

☐ **S'il y a d'autres priorités, cochez la case et utilisez l'imprimé «Suite»**

5 DEMANDEUR (Cochez l'une des 2 cases) ☐ **Personne morale** ☐ **Personne physique**

Nom ou dénomination sociale		FRANCE TELECOM
Prénoms		
Forme juridique		Société Anonyme
N° SIREN		3 8 0 1 2 9 8 6 6
Code APE-NAF		
Domicile ou siège	Rue	6, place d'Alleray
	Code postal et ville	7 5 0 1 5 PARIS
	Pays	FRANCE
Nationalité		Française
N° de téléphone *(facultatif)*		N° de télécopie *(facultatif)*
Adresse électronique *(facultatif)*		

☐ **S'il y a plus d'un demandeur, cochez la case et utilisez l'imprimé «Suite»**

Remplir impérativement la 2ème page

INPI
INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

REQUÊTE EN DÉLIVRANCE
page 2/2

BR2

Réservé à l'INPI
REMISE DES PIÈCES
DATE 24 JAN 2003
LIEU 75 INPI PARIS F
N° D'ENREGISTREMENT NATIONAL ATTRIBUÉ PAR L'INPI 0301108

DB 540 W / 210502

6 MANDATAIRE *(s'il y a lieu)*		
Nom		JEUNE
Prénom		Pascale
Cabinet ou Société		FRANCE TELECOM R&D/VAT/PI
N °de pouvoir permanent et/ou de lien contractuel		PG 8611
Adresse	Rue	38-40, rue du Général Leclerc
	Code postal et ville	9 2 7 9 4 ISSY MOULINEAUX
	Pays	FRANCE
N° de téléphone *(facultatif)*		01 45 29 65 78
N° de télécopie *(facultatif)*		01 45 29 65 60
Adresse électronique *(facultatif)*		
7 INVENTEUR (S)		**Les inventeurs sont nécessairement des personnes physiques**
Les demandeurs et les inventeurs sont les mêmes personnes		☐ Oui ☒ Non : **Dans ce cas remplir le formulaire de Désignation d'Inventeur(s)**
8 RAPPORT DE RECHERCHE		**Uniquement pour une demande de brevet (y compris division et transformation)**
Établissement immédiat ou établissement différé		☒ ☐
Paiement échelonné de la redevance *(en deux versements)*		**Uniquement pour les personnes physiques effectuant elles-mêmes leur propre dépôt** ☐ Oui ☐ Non
9 RÉDUCTION DU TAUX DES REDEVANCES		**Uniquement pour les personnes physiques** ☐ Requise pour la première fois pour cette invention *(joindre un avis de non-imposition)* ☐ Obtenue antérieurement à ce dépôt pour cette invention *(joindre une copie de la décision d'admission à l'assistance gratuite ou indiquer sa référence)* : **AG**
10 SÉQUENCES DE NUCLEOTIDES ET/OU D'ACIDES AMINÉS		☐ Cochez la case si la description contient une liste de séquences
Le support électronique de données est joint		☐
La déclaration de conformité de la liste de séquences sur support papier avec le support électronique de données est jointe		☐
Si vous avez utilisé l'imprimé «Suite», indiquez le nombre de pages jointes		

11 SIGNATURE DU DEMANDEUR OU DU MANDATAIRE (Nom et qualité du signataire)

JEUNE Pascale
Mandataire par pouvoir PG 8611

VISA DE LA PRÉFECTURE OU DE L'INPI

La loi n°78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés s'applique aux réponses faites à ce formulaire.
Elle garantit un droit d'accès et de rectification pour les données vous concernant auprès de l'INPI.

PROCEDE CRYPTOGRAPHIQUE A CLE PUBLIQUE POUR LA PROTECTION D'UNE PUCE ELECTRONIQUE CONTRE LA FRAUDE

La présente invention se rapporte au domaine de la cryptographie. En particulier, l'invention se rapporte aux procédés cryptographiques de protection contre la fraude d'une puce électronique dans des transactions entre une application et la puce.

L'invention trouve une application très avantageuse en ce qu'elle permet de protéger contre la fraude des puces à circuit intégré à logique câblée ou à microprocesseur, notamment les puces qui équipent les cartes prépayées utilisées dans des transactions diverses telles que l'établissement de communications téléphoniques, le paiement d'objets dans un distributeur automatique, la location d'emplacements de stationnement à partir d'un parcmètre, le paiement d'un service comme un transport public ou comme la mise à disposition d'infrastructures (péage, musée, bibliothèque,...).

Actuellement, les cartes prépayées sont susceptibles de subir différents types de fraude. Un premier type de fraude consiste à dupliquer sans autorisation la carte, le terme clonage étant souvent utilisé pour caractériser cette opération. Un deuxième type de fraude consiste à modifier les données attachées à une carte, en particulier le montant du crédit inscrit dans la carte. Pour lutter contre ces fraudes il est fait appel à la cryptographie, d'une part pour assurer l'authentification de la carte au moyen d'une authentification et/ou pour assurer l'authentification des données au moyen d'une signature numérique et, d'autre part pour assurer le cas échéant la confidentialité des données au moyen d'un chiffrement. La cryptographie met en jeu deux entités, qui sont dans le cas de l'authentification un vérificateur et un objet à vérifier, et elle peut être soit symétrique, soit asymétrique. Lorsqu'elle est symétrique (ou "à clé secrète", les deux termes étant synonymes), les deux entités partagent exactement la même information, en particulier une clé secrète. Lorsqu'elle est asymétrique (ou "à clé publique", les deux termes étant synonymes) une des deux entités possède une paire de clés dont l'une est secrète et l'autre est publique ; il n'y a pas de clé secrète partagée. Dans de nombreux systèmes, notamment lorsque la puce est de type "à logique câblée", seule la cryptographie symétrique est mise en œuvre avec des cartes prépayées, car la cryptographie asymétrique reste lente et coûteuse. Les premiers mécanismes d'authentification développés en cryptographie symétrique consistent à calculer une fois pour toutes une valeur d'authentification, différente pour chaque carte, à la stocker dans la mémoire de la carte, à la lire à chaque transaction et à la vérifier en interrogeant

une application du réseau supportant la transaction où les valeurs d'authentification déjà attribuées sont soit stockées soit recalculées. Ces mécanismes assurent une protection insuffisante parce que la valeur d'authentification peut être espionnée, reproduite et rejouée frauduleusement étant donné qu'elle est toujours la même pour une carte donnée, permettant ainsi de réaliser un clone de cette carte. Pour lutter contre les clones, les mécanismes d'authentification passifs de cartes sont remplacés par des mécanismes d'authentification actifs qui peuvent en outre assurer l'intégrité des données.

Le principe général des mécanismes d'authentification actifs symétriques est le suivant : lors d'une authentification, la puce électronique et l'application calculent une valeur d'authentification qui est le résultat d'une fonction appliquée à une liste d'arguments déterminée à chaque authentification ; la liste d'arguments pouvant comprendre un aléa, l'aléa étant une donnée déterminée par l'application à chaque authentification, une donnée contenue dans la puce électronique et une clé secrète connue de la puce électronique et de l'application. Lorsque la valeur d'authentification calculée par la puce électronique est identique à la valeur d'authentification calculée par l'application, la puce électronique est jugée authentique et la transaction entre la puce électronique et l'application est autorisée.

De tels mécanismes d'authentification sont largement connus mais la plupart exigent des capacités de calcul au moins égales à celles dont dispose un microprocesseur. Ces mécanismes conviennent donc aux cartes à microprocesseur, mais rarement aux puces à logique câblée, lesquelles disposent de moyens de calcul beaucoup plus rudimentaires.

Un premier pas a été effectué lorsque des mécanismes actifs d'authentification symétriques ont pu être intégrés dans des puces à logique câblée. Par exemple, la demande de brevet française publiée le 27.12.2002 sous le numéro FR 2826531 décrit un procédé permettant de spécifier de tels mécanismes. On notera que la valeur d'authentification produite par ces mécanismes peut aussi, comme l'enseigne la demande de brevet française précédente, être interprétée comme une séquence de bits pseudo-aléatoires et, en faisant varier au moins l'un des paramètres d'entrée, le procédé de calcul de la valeur d'authentification devient alors un procédé de génération de bits pseudo-aléatoires.

Cependant, les mécanismes à clé secrète imposent que le dispositif de vérification, en charge de l'authentification de la puce, tel que ceux présents dans un téléphone public, un terminal de paiement électronique, ou encore un portillon de

transport en commun, connaisse la clé secrète détenue par ladite puce. Il en découle un inconvénient majeur, à savoir que, si l'on souhaite que ledit dispositif puisse authentifier n'importe quelle puce émise en relation avec l'application, soit il doit stocker les clés secrètes de toutes les puces, soit il doit stocker une clé de base, appelée aussi clé-mère ou clé-maître, permettant de retrouver la clé secrète de n'importe quelle puce. Dans les deux cas, chacun de ces dispositifs stocke suffisamment d'information pour pouvoir retrouver les clés secrètes de toutes les puces émises, et stocke donc suffisamment d'information pour pouvoir fabriquer des clones de n'importe laquelle d'entre elles. Il s'ensuit qu'une intrusion réussie contre n'importe lequel des dispositifs de vérification anéantirait la sécurité de l'application dans son ensemble.

Il existe donc un besoin impérieux de pouvoir intégrer un mécanisme actif d'authentification à clé publique dans une puce à logique câblée, notamment dans les applications déployant un grand nombre de puces, ce qui est généralement le cas des applications utilisant des puces à logique câblée, car elles sont très bon marché. Or un tel mécanisme n'existe pas à l'heure actuelle. La raison en est que les mécanismes à clé publique requièrent généralement de nombreuses opérations portant sur de grands nombres, et donc qu'ils sont inappropriés à une intégration dans des puces à logique câblée, dans lesquelles la surface de silicium est extrêmement réduite, et dont la logique de calcul se réduit au câblage d'opérations extrêmement élémentaires. Ces opérations élémentaires sont effectuées généralement en série, en ce sens que les opérandes sont introduites séquentiellement bit après bit, et que cette introduction modifie progressivement l'état d'un registre interne dont la valeur finale sert de base au calcul du résultat de la fonction.

La présente invention se rapporte aux mécanismes actifs d'authentification à clé publique qui peuvent être mis en œuvre dans une carte à logique câblée.

Plus précisément, la présente invention porte sur un procédé cryptographique asymétrique de protection contre la fraude de la puce électronique, dans des transactions entre une application et la puce électronique, plus particulièrement adaptés aux puces à logique câblée et plus particulièrement destinés à mettre en place un mécanisme d'authentification, qui soit dépourvu des inconvénients de la cryptographie symétrique mentionnés ci-dessus, de manière à renforcer la sécurité de l'application dans son ensemble, et en particulier de rendre la création de clones plus ardue.

A cette fin l'invention a pour objet un procédé cryptographique asymétrique de protection contre la fraude d'une puce électronique, selon la revendication 1.

L'invention a en outre pour objet un dispositif à puce électronique selon la revendication 32.

L'invention a en outre pour objet un dispositif de vérification selon la revendication 33.

Un procédé selon l'invention a pour avantage de permettre la production d'une valeur d'authentification V vérifiable exclusivement au moyen de paramètres publics, tout en étant produite exclusivement par des fonctions série, c'est-à-dire des fonctions traitant séquentiellement les bits des paramètres qui en constituent les entrées.

Les paramètres d'entrée du procédé cryptographique et du dispositif sont traités dans la fonction série qui fournit en sortie une donnée dépendant de tout ou partie des paramètres d'entrée.

Les paramètres d'entrée du procédé et du dispositif appartiennent à une liste qui comprend, dans le cas de la mise en œuvre d'un mécanisme d'authentification, au moins un identifiant I, une clé s privée secrète, une clé publique p correspondant à la clé s privée, d'un certificat de cette clé publique, d'un deuxième aléa t fourni par le dispositif de vérification.

Le générateur pseudo-aléatoire série permettant de calculer l'aléa r peut avantageusement reposer sur un procédé d'authentification symétrique du type de ceux décrits dans la demande de brevet française publiée le 27.12.2002 sous le numéro FR 2826531 sus-mentionnée. Ainsi, si on désigne par f(K,M) la fonction de calcul d'un tel procédé, notation dans laquelle K désigne la clé secrète symétrique et M désigne l'ensemble des autres opérandes de la fonction f, alors l'aléa r peut être produit par application répétée de la fonction f à des valeurs différentes de M tout en conservant la même valeur de K. A titre d'exemple, si la taille de la valeur de sortie z de f est égale à k bits et si la taille de l'aléa r est égale à 16k bits, le premier aléa r utilisé lors de la première authentification de la puce peut être choisi égal à la concaténation des seize valeurs de sortie $f(K, M_1)$, $f(K, M_2)$,..., $f(K, M_{16})$; le second aléa peut être choisi égal à la concaténation des seize valeurs de sortie $f(K, M_{17})$, $f(K, M_{18})$,..., $f(K, M_{32})$ etc., toutes les valeurs M_i étant distinctes les une des autres (typiquement, la valeur de M_{i+1} est obtenue en incrémentant la valeur de M_i). De nombreuses autres façons d'utiliser le procédé d'authentification à des fins de générateur pseudo-aléatoire sont possibles.

La fonction série contient des additions, des soustractions et des décalages à gauche ou à droite. En effet, ces opérations peuvent être très facilement réalisées de façon séquentielle.

D'autres caractéristiques et avantages de l'invention apparaîtront lors de la description qui suit faite en regard de dessins annexés de modes particuliers de réalisation donnés à titre d'exemples non limitatifs.

La figure 1 est un organigramme d'un procédé selon l'invention.

La figure 2 est un schéma d'un dispositif à puce électronique selon l'invention.

La figure 3 est un schéma d'un exemple d'un générateur pseudo-aléatoire d'un dispositif à puce électronique selon l'invention.

La figure 4 est un schéma d'un exemple d'un moyen de mise en œuvre d'une fonction série d'un dispositif à puce électronique selon l'invention.

La figure 1 représente un organigramme d'un procédé cryptographique asymétrique de protection contre la fraude d'une puce électronique selon l'invention, dans des transactions entre une application et la puce électronique.

Le procédé consiste à calculer dans la puce électronique une valeur d'authentification à partir de paramètres d'entrée.

Dans une première étape le procédé consiste à produire 1 par la puce un nombre pseudo-aléatoire dit aléa r au moyen d'un générateur pseudo-aléatoire série inclus dans la puce. L'aléa r est propre à la transaction.

Dans une deuxième étape le procédé consiste à transmettre 2 de la puce à l'application un paramètre x. Ce paramètre x est calculé préalablement à la transaction par l'application et stocké en mémoire de données de la puce. Ce paramètre x est relié à l'aléa r par une relation mathématique. L'application calcule au moins un paramètre x, avantageusement elle en calcule plusieurs. Selon un cas particulier de mise en œuvre, ces paramètres x sont le résultat d'une fonction mathématique appliquée à des valeurs prises successivement dans un ensemble donné pour une puce donnée. Cet ensemble est tel que les différentes valeurs de l'aléa r générées par la puce soient comprises dans cet ensemble.

Ainsi la fonction mathématique reliant les aléas r et les paramètres x consiste typiquement en une exponentielle dans un ensemble G muni d'une opération ayant au moins pour propriété d'être associative et notée sous la forme d'une multiplication, c'est-à-dire que la fonction est $x = g^r$, où r désigne un entier et g désigne un élément dudit ensemble G choisi préalablement par l'application.

r est un nombre pseudo-aléatoire, différent pour chaque puce et pour chaque authentification. Il est calculé à deux reprises : la première fois par l'application, la deuxième fois par la puce elle-même. Après avoir calculé r, l'application calcule le x correspondant.. L'application stocke ensuite au moins une valeur de x dans la puce au

moment de la personnalisation de cette dernière. Avantageusement, l'application stocke plusieurs valeurs de x. Comme l'application et la puce doivent produire la même valeur de *r*, il est bien sûr impératif que le générateur pseudo-aléatoire de l'application et celui de la puce soient strictement identiques.

g peut être avantageusement le même pour toutes les puces électroniques liées à l'application ou peut être propre à la puce. Dans ce dernier cas, g fait partie intégrante de la clé p publique de la puce électronique. Des exemples typiques d'ensembles G sont, n désignant un entier positif quelconque, le groupe Z_n^* des entiers positifs ou nuls inférieurs à n et premiers avec n, ou bien encore, toute courbe elliptique construite sur un corps fini.

Dans une troisième étape, le procédé consiste à calculer 3 par la puce un paramètre y au moyen d'une fonction série ayant pour paramètres d'entrée au moins l'aléa r propre à la transaction et une clé s privée secrète appartenant à une paire de clés asymétrique (s, p), ce paramètre y constituant tout ou partie de la valeur d'authentification V. La fonction série consiste en une fonction arithmétique.

Dans une quatrième étape, le procédé consiste à transmettre 4 la valeur d'authentification V de la puce à l'application.

Dans une cinquième étape, le procédé consiste à vérifier 5 par l'application ladite valeur d'authentification V au moyen d'une fonction de vérification dont les paramètres d'entrées consistent exclusivement en des paramètres publics, contenant au moins la clé p publique.

La figure 2 représente schématiquement un dispositif à puce électronique selon l'invention. Le dispositif à puce électronique permet la mise en œuvre d'un procédé cryptographique asymétrique de protection contre la fraude de la puce électronique selon l'invention, dans des transactions entre une application et la puce électronique, consistant à calculer par la puce électronique une valeur V d'authentification à partir de paramètres d'entrée.

Le dispositif 6 comprend :

- un générateur 7 pseudo-aléatoire série produisant un aléa r propre à la transaction,

- un premier moyen 8 de mémorisation dans lequel sont stockés un ou plusieurs paramètres x calculés préalablement à la transaction par l'application, chacun de ces paramètres x étant relié par une même relation mathématique à une valeur de l'aléa r comprise dans un ensemble de valeurs pouvant être produites par le générateur pseudo-aléatoire série,

-un premier moyen 9 de sortie du paramètre x relié à l'aléa r propre à la transaction,

- un moyen 10 de mise en œuvre d'une fonction série ayant pour paramètres d'entrées au moins l'aléa r propre à la transaction et une clé s privée appartenant à une paire de clés asymétrique (s, p), ce paramètre y constituant tout ou partie de la valeur d'authentification V,

- un second moyen 9 de sortie de la valeur V d'authentification après constitution de cette valeur à partir d'au moins le paramètre y.

Le générateur 7 pseudo-aléatoire série repose, dans le cas de l'exemple retenu et décrit en regard de la figure 2, sur un procédé d'authentification symétrique du type de ceux décrits dans la demande de brevet française publiée le 27.12.2002 sous le numéro FR 2826531 sus-mentionnée. Ainsi, si on désigne par f(K,M) la fonction de calcul d'un tel procédé, notation dans laquelle K désigne la clé secrète symétrique et M désigne l'ensemble des autres opérandes de la fonction f, alors l'aléa r est produit par application répétée de la fonction f à des valeurs différentes de M tout en conservant la même valeur de K. A titre d'exemple, si la taille de la valeur de sortie z de f est égale à k bits et si la taille de l'aléa r est égale à 16k bits, le premier aléa r utilisé lors de la première authentification de la puce est choisi égal à la concaténation des seize valeurs de sortie $f(K, M_1)$, $f(K, M_2)$,..., $f(K, M_{16})$; le second aléa choisi est égal à la concaténation des seize valeurs de sortie $f(K, M_{17})$, $f(K, M_{18})$,..., $f(K, M_{32})$ etc, toutes les valeurs M_i étant distinctes les une des autres.

La figure 3 schématise un tel générateur 6 pseudo-aléatoire série. Ce générateur comprend des moyens 12 de mélange de tout ou partie des paramètres d'entrée pour fournir en sortie une donnée $E' = (e'_1, e'_2, ...e'_n, ..., e'_N)$ résultat du mélange, un automate 13 à états finis qui passe d'un état ancien à un état nouveau selon une fonction dépendant au moins de l'état ancien et d'une valeur de la suite de bits $(e'_1, e'_2, ...e'_n, ..., e'_N)$, un moyen 14 de sortie pour calculer la valeur z à partir d'arguments d'entrée comprenant au moins un état de l'automate, puis pour déterminer la valeur de l'aléa r choisi en effectuant la concaténation de seize valeurs de sortie $f(K, M_1)$, $f(K, M_2)$,..., $f(K, M_{16})$ successives. Les paramètres d'entrée des moyens 12 de mélange peuvent être pris dans une liste non exhaustive qui comprend : une clé secrète K, des données D internes à la puce, l'adresse mémoire des données D, des données D' externes à la puce, un aléa R' fourni par l'application.

Les moyens 12 de mélange mettent en œuvre une fonction de mélange MIX qui peut être une fonction linéaire ou non-linéaire des données d'entrée.

Un premier exemple de fonction linéaire consiste à effectuer le produit scalaire entre les données d'entrée.

Selon un autre exemple de mise en œuvre d'une fonction linéaire, les moyens de mélange comprennent un registre à décalage à rétroaction linéaire dans lequel les bits des paramètres d'entrée sont entrés successivement et influent sur l'état initial du registre et/ou sur la valeur des bits de rétroaction.

Selon un exemple de mise en œuvre d'une fonction non-linéaire, les moyens de mélange comprennent un registre à décalage à rétroaction non linéaire, dans lequel les bits des paramètres d'entrée sont entrés successivement. La valeur de sortie S' peut être constituée d'un ou plusieurs bits extraits du contenu de ce registre.

Un premier exemple d'automate 13 AUT consiste à utiliser un circuit booléen. C'est-à-dire un circuit qui, par exemple à un vecteur de k+1 bits $(A_1,A_2,\ldots A_{k+1})$ associe un vecteur de k bits $(A'_1,A'_2,\ldots A'_k)$, où chaque bit A'_i est obtenu à partir des bits $(A_1,A_2,\ldots A_{k+1})$ à l'aide d'opérations élémentaires telles que OU exclusif, OU (inclusif), ET, NON et où $(A_1,A_2,\ldots A_k)$ représente l'état ancien de l'automate. L'automate possède un état interne de k bits $(A_1,A_2,\ldots A_k)$ et présente en sortie un nouvel état $(A'_1,A'_2,\ldots A'_k)$, à chaque fois qu'un nouveau vecteur $(A_1,A_2,\ldots A_k$, S'e') est présent en entrée du circuit booléen, le nouveau vecteur étant constitué de l'état interne et de la sortie de la fonction de mélange MIX.

Un second exemple d'automate 13 consiste à utiliser des transformations de bits définies par des tableaux de nombres. Dans le cas où k=8, il est par exemple possible de diviser l'octet $(A_1,A_2,\ldots A_8)$ en deux quartets (A_1,A_2,A_3,A_4) et (A_5,A_6,A_7,A_8), puis d'appliquer à chaque quartet une transformation T si le bit de sortie E'e' de la fonction de mélange vaut zéro, ou une transformation U si E'e' vaut un. La transformation T est définie par un tableau qui associe à chaque valeur de quartet (a,b,c,d) une valeur de quartet (a',b',c',d'). De même pour U.

Lorsque toutes les valeurs d'entrée ont été prises en compte, l'automate 13 est dans un certain état final $(F_1,F_2,\ldots F_k)$.

Le moyen 14 de sortie du générateur pseudo-aléatoire série met typiquement en œuvre une fonction de sortie qui est la fonction identité appliquée à l'état final de l'automate et une opération de concaténation. Ce moyen est par exemple une zone mémoire de taille égale à la taille de l'aléa r, 16k bits.

Le premier moyen 8 de mémorisation dans lequel sont stockés un ou plusieurs paramètres x consiste typiquement en une mémoire du type non volatile pouvant éventuellement être ré-écrite. Les paramètres x sont écrits dans la mémoire avant commercialisation de la puce électronique. La valeur de l'aléa r intervenant dans le calcul du paramètre x doit être choisie de façon telle que la puce soit capable de recalculer cette valeur à l'identique. Dans l'exemple de générateur pseudo-aléatoire série décrit en regard de la figure 2, cette condition implique que la clé secrète K soit partagée par la puce et par l'application. Ainsi, avant la mise en circulation de la puce, l'application calcule un certain nombre de valeurs de x par application répétée du procédé d'authentification dont la fonction de calcul a été notée f ci-dessus, et stocke ces valeurs dans la mémoire de données de la puce. A chaque authentification, la puce recalcule l'aléa r et lit en mémoire de données la valeur du paramètre x qui lui correspond. Dans l'exemple de générateur pseudo-aléatoire série décrit en regard de la figure 2, la correspondance entre r et x est typiquement établie en choisissant pour valeur de M_1 une information permettant de déterminer l'adresse de la valeur de x correspondant à cette valeur de r, la valeur de M_{i+1} pour i supérieur ou égal à 0 étant obtenue en incrémentant la valeur de M_i.

Afin d'économiser de la place en mémoire, le paramètre x peut être avantageusement choisi égal à l'image par une fonction de hachage h de l'élément g^r (ainsi éventuellement qu'à d'autres éléments, tels que des données d'application), plutôt qu'à cet élément lui-même, c'est-à-dire que l'on a : $x = h(g^r, D)$, où D désigne un champ optionnel contenant par exemple des données liée à l'application. Par exemple, D désigne un montant en euro décidé par l'application. Dans ce cas, chaque coupon représente une pièce de monnaie électronique, et chaque authentification représente la dépense d'une telle pièce.

Le premier moyen 9 de sortie du paramètre x relié à l'aléa r propre à la transaction est typiquement un buffer d'entrées/sorties.

Un exemple de moyen 10 de mise en œuvre d'une fonction série est décrit en regard de la figure 4. La fonction série a pour paramètres d'entrée l'aléa r et une clé s privée secrète appartenant à une paire de clés asymétrique (s, p). La clé p est publique.

Le moyen 10 se compose d'un additionneur de bits avec calcul et prise en compte d'une retenue.

La valeur du bit courant r_i de r est capturée dans un premier registre 15, la valeur du bit courant s_i de s est capturée dans un second registre 16. Un troisième

registre 17 capture la retenue c_i qui résulte des additions de bits qui ont précédé. Enfin, un quatrième registre 18 capture le bit y_i obtenu après addition des valeurs des bits courants r_i et s_i avec la retenue obtenue lors de l'addition précédente, cette retenue correspondant au contenu du troisième registre 17. La retenue c_i résulte de la prise en compte de la retenue générée lors de l'addition des bits précédents, sortie du composant 19 ET dont les entrées sont les sorties des deux premiers registres 15, 16, et de la retenue générée lors de l'addition des bits courants, sortie du composant 20 ET dont les entrées sont les valeurs des bits courants r_i et s_i. Un composant 21 ET intermédiaire génère une retenue lorsqu'il y a une retenue générée lors de l'addition des bits précédents et lorsqu'un seul des bits courants est à un, sortie du composant 22 ou exclusif dont les entrées sont les valeurs des bits courants.

La retenue c_i est donc un 23 OU entre la sortie du composant 21 ET intermédiaire et du composant 20 ET dont les entrées sont les valeurs des bits courants r_i et s_i. Cette retenue c_i est capturée dans le troisième registre 17 pour être prise en compte lors de l'addition des bits suivants de r_i et s_i.

Le bit y_i résulte de l'addition des valeurs des bits courants r_i et s_i, sortie du composant 22 OU exclusif dont les entrées sont les valeurs des bits courants r_i et s_i, avec la valeur de la retenue, sortie du composant 24 OU exclusif dont les entrées sont la sortie du composant 22 OU exclusif précédent et la sortie du troisième registre 17.

Les sorties des registres 15, 16, 17, 18 sont initialisées à 0.

Ceci donne en final : $y_i = r_i + s_i + c_i (\mathrm{mod}\, 2)$ et $c_{i+1} = r_i + s_i + c_i (\mathrm{div} 2)$, où c_0 est choisi égal à 0.

Selon une application particulière, la fonction série a en outre pour paramètre d'entrée un aléa t fournit par l'application.

Après que la puce a produit un aléa r selon le procédé décrit en regard de la figure 2, puis lu la valeur du paramètre x qui correspond à la valeur dudit aléa dans sa mémoire de données (par exemple à travers la fonction $x = g^r$), elle envoie à l'application la valeur de x, après quoi l'application envoie à la puce un aléa t dont la taille est réduite à 1 bit.

Deux cas se présentent alors. Si la valeur de t est égale à 0, la puce choisit y = r. Si la valeur de t est égale à 1, la puce choisit y = r + s. L'implantation de ce choix est connue de l'homme du métier et n'est donc pas détaillée.

La valeur d'authentification V est prise égale à y, et est envoyée à l'application.

La vérification consiste à tester l'équation : $g^y = x$ si t est égal à 0, ou bien $g^y = xp$, si t est égal à 1, où p est la clé publique de la puce correspondant à sa clé

secrète s, définie par la fonction $p = g^s$. Si ces paramètres sont choisis suffisamment grands, il est infaisable de retrouver s à partir de g et p selon l'hypothèse dite du logarithme discret, qui est une hypothèse aujourd'hui communément admise.

Selon une application particulière, une fonction de hachage h peut être utilisée pour le calcul de x. Dans ce cas, l'équation de vérification devient : $h(g^y, D) = x$ si t est égal à 0 ou bien $h(g^y / p, D) = x$ si t est égal à 1. Pour éviter toute division dans l'équation de vérification, il est aussi possible de choisir y = r - s plutôt que y = r + s, auquel cas la seconde équation de vérification devient : $h(g^y.p, D) = x$. Une autre possibilité est de choisir $p = g^{-s}$ plutôt que $p = g^s$, qui conduit aux équations de vérification suivantes : $h(g^y, D) = x$ et $h(g^y.p, D) = x$.

Dans les mises en œuvre précédemment décrites, toute autre puce que celle connaissant la valeur secrète s a au plus une chance sur deux de fournir une valeur d'authentification qui soit reconnue comme valide par l'application. Ceci permet déjà d'établir une distinction entre une puce authentique et un clone, mais cette distinction reste insuffisante dans la plupart des cas réels d'application.

Pour diminuer notablement les chances de succès d'un clone, une solution consiste à augmenter le nombre de bits m de l'aléa t. Par exemple, l'aléa t est choisi égal à une chaîne de 64 bits (t_{63}, t_{62},..., t_0) dont un seul des bits est égal à 1. Soit i l'unique indice tel que t_i est égal à 1, alors la valeur de y est choisie égale à : $y = r + 2^i s$. Cette valeur est très facile à calculer séquentiellement puisque cela revient à effectuer une addition entre r et l'entier obtenu en décalant s de i bits vers la gauche (si les poids forts se trouvent à gauche). L'équation de vérification est alors : $g^y = xp^{2^i}$. Dans ces conditions, toute autre puce que celle connaissant la valeur secrète s, a au plus une chance sur 64 de fournir une valeur d'authentification qui soit reconnue comme valide par l'authentification.

Selon une application particulière, une fonction de hachage h peut être utilisée pour le calcul de x. Dans ce cas, l'équation de vérification devient : $h(g^y / p^{2^i}, D) = x$. Pour éviter toute division dans l'équation de vérification, il est aussi possible de choisir $y = r - 2^i s$ plutôt que $y = r + 2^i s$, auquel cas la seconde équation de vérification devient : $h(g^y.p^{2^i}, D) = x$. Une autre possibilité est de choisir $p = g^{-s}$ plutôt que $p = g^s$, qui conduit à l'équation de vérification suivante : $h(g^y.p^{2^i}, D) = x$.

Pour cette solution décrite, il revient au même du point de vue de la sécurité de choisir pour valeur de t un entier compris entre 0 et m-1 à la place de la chaîne t telle

que définie ci-dessus, y étant alors pris égal à : $y = r + 2^t s$ et l'équation de vérification étant : $g^y = xp^{2^t}$.

Selon une application particulière, une fonction de hachage h peut être utilisée pour le calcul de x. Dans ce cas, l'équation de vérification devient : $h(g^y / p^{2^t}, D) = x$. Pour éviter toute division dans l'équation de vérification, il est aussi possible de choisir $y = r - 2^t s$ plutôt que $y = r + 2^t s$, auquel cas la seconde équation de vérification devient : $h(g^y . p^{2^t}, D) = x$. Une autre possibilité est de choisir $p = g^{-s}$ plutôt que $p = g^s$, qui conduit à l'équation de vérification suivante : $h(g^y . p^{2^t}, D) = x$.

Il revient encore au même du point de vue de la sécurité de choisir pour valeur de t un entier compris entre 0 et m-1 à la place de la chaîne t telle que définie ci-dessus, y étant alors pris égal à : $y = r + ts$ et l'équation de vérification étant : $g^y = xp^t$.

Selon une application particulière, une fonction de hachage h peut être utilisée pour le calcul de x. Dans ce cas, l'équation de vérification devient : $h(g^y / p^t, D) = x$. Pour éviter toute division dans l'équation de vérification, il est aussi possible de choisir $y = r - ts$ plutôt que $y = r + ts$, auquel cas la seconde équation de vérification devient : $h(g^y . p^t, D) = x$. Une autre possibilité est de choisir $p = g^{-s}$ plutôt que $p = g^s$, qui conduit à l'équation de vérification suivante : $h(g^y . p^t, D) = x$.

L'aléa t peut bien entendu prendre d'autres valeurs.

Le second moyen 9 de sortie de la valeur V d'authentification met typiquement en œuvre une fonction de sortie qui est la fonction identité appliquée au paramètre y. Ce moyen 9 est par exemple une zone mémoire de taille égale à la taille du paramètre y.

Lors de transactions entre une application et une puce électronique, l'application et la puce mettent en œuvre un procédé cryptographique asymétrique de protection contre la fraude de la puce électronique selon l'invention. Lors de cette mise en œuvre, l'application fait appel à un dispositif de vérification selon l'invention pour authentifier la puce. Le dispositif comprend un moyen de mise en œuvre de la fonction de vérification d'un procédé selon l'invention. Ce moyen vérifie la valeur V d'authentification calculée par la puce électronique en utilisant des paramètres

exclusivement publics qui comprennent au moins la clé p publique liée à la clé s secrète de la puce.

Selon un des modes de réalisation précédemment décrits d'un procédé selon l'invention, le dispositif de vérification compare le résultat (g^y) fourni par la fonction mathématique appliquée à la valeur d'authentification V à une des valeurs suivantes : la valeur x, le produit (xp) de la valeur x avec la clé p publique de la puce correspondant à sa clé s secrète, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par la fonction $p = g^s$.

Le moyen est typiquement un calculateur.

REVENDICATIONS

1. Procédé cryptographique asymétrique de protection contre la fraude d'une puce électronique, dans des transactions entre une application et la puce électronique, consistant à calculer dans la puce électronique une valeur V d'authentification à partir de paramètres d'entrée, ledit procédé comprenant les étapes qui consistent :

- à produire (1) par la puce un nombre pseudo-aléatoire dit aléa r propre à la transaction au moyen d'un générateur pseudo-aléatoire série inclus dans la puce,
- à transmettre (2) de la puce à l'application un paramètre x calculé par l'application préalablement à la transaction, relié à l'aléa r par une relation mathématique et stocké en mémoire de données de la puce,
- à calculer (3) par la puce un paramètre y au moyen d'une fonction série ayant pour paramètres d'entrées au moins l'aléa r propre à la transaction et une clé s privée appartenant à une paire de clés asymétrique (s, p), ce paramètre y constituant tout ou partie de la valeur d'authentification V,
- à transmettre (4) la valeur d'authentification V de la puce à l'application et,
- à vérifier (5) par l'application ladite valeur d'authentification V au moyen d'une fonction de vérification dont les paramètres d'entrées consistent exclusivement en des paramètres publics, contenant au moins la clé p publique.

2. Procédé cryptographique asymétrique de protection contre la fraude d'une puce électronique selon la revendication 1 dans lequel la production de l'aléa r propre à la transaction consiste :

- à mélanger tout ou partie de paramètres d'entrée au moyen d'une fonction (12) de mélange et à fournir en sortie de la fonction de mélange une suite de bits,
- à effectuer le changement d'état d'un automate (13) à états finis en le faisant passer d'un état ancien à un état nouveau selon une fonction dépendant au moins de l'état ancien et d'une valeur de la suite de bits,
- à déterminer une suite de bits aléatoires pour former tout ou partie de l'aléa r au moyen d'une fonction (14) de sortie ayant pour argument d'entrée au moins un état de l'automate.

3. Procédé selon la revendication 2, dans lequel l'un des paramètres d'entrée est constitué d'une clé secrète K partagée entre la puce et l'application, stockée dans une zone-mémoire protégée de la puce.

4. Procédé selon la revendication 1, dans lequel la relation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative.

5. Procédé selon la revendication 4, dans lequel l'ensemble G est le groupe Z_n^* des entiers positifs ou nuls inférieurs à n et premiers avec n.

6. Procédé selon la revendication 4, dans lequel l'ensemble G est toute courbe elliptique construite sur tout corps fini quelconque.

7. Procédé selon la revendication 1, dans lequel la fonction série est une fonction arithmétique effectuant des opérations prises dans une liste qui comprend des additions, des soustractions et des décalages à gauche ou à droite.

8. Procédé selon la revendication 7, dans lequel la fonction arithmétique effectue uniquement des additions.

9. Procédé selon la revendication 7, dans lequel la fonction arithmétique effectue uniquement des soustractions.

10. Procédé selon la revendication 7, dans lequel la fonction arithmétique a en outre pour arguments d'entrée des paramètres d'entrée et consiste à effectuer une des opérations suivantes : $y = r$ et $y = r + s$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série.

11. Procédé selon la revendication 10, dans lequel la relation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel la fonction de vérification compare le résultat fourni par la fonction appliquée à la valeur d'authentification V à une des valeurs suivantes : la valeur x, le produit de la valeur x avec la clé p publique de la puce correspondant à sa clé s secrète, en fonction de la

valeur du paramètre t, ce qui revient à tester une des équations suivantes : $g^y = x$ et $g^y = xp$, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par la fonction $p = g^s$.

12. Procédé selon la revendication 7, dans lequel la fonction arithmétique a en outre pour arguments d'entrée des paramètres d'entrée et consiste à effectuer une des opérations suivantes : $y = r$ et $y = r - s$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série.

13. Procédé selon la revendication 12, dans lequel l'équation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel la fonction de vérification compare le résultat fourni par l'équation mathématique appliquée à la valeur d'authentification V à une des valeurs suivantes : la valeur x, le produit de la valeur x avec la clé p publique de la puce correspondant à sa clé s secrète, en fonction de la valeur du paramètre t, ce qui revient à tester une des équations suivantes : $g^y = x$ et $g^y.p = x$, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par l'équation $p = g^s$.

14. Procédé selon la revendication 7, dans lequel la fonction arithmétique a en outre pour arguments d'entrée des paramètres d'entrée et consiste à effectuer l'opération suivante : $y = r + 2^i s$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série, ce paramètre t consistant en une chaîne de m bits ($t_{m-1},\ldots,t_0$) dont un seul des bits est égal à 1, le bit t_i, m étant un entier naturel.

15. Procédé selon la revendication 14, dans lequel la relation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel la fonction de vérification consiste à tester l'équation suivante : $g^y = xp^{2^i}$, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par la fonction $p = g^s$.

16. Procédé selon la revendication 7, dans lequel la fonction arithmétique a en outre pour arguments d'entrée des paramètres d'entrée et consiste à effectuer l'opération suivante : $y = r + 2^t s$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série.

17. Procédé selon la revendication 16, dans lequel la relation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel la fonction de vérification consiste à tester l'équation suivante : $g^y = xp^{2^t}$, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par la fonction $p = g^s$.

18. Procédé selon la revendication 7, dans lequel la fonction arithmétique a en outre pour arguments d'entrée des paramètres d'entrée et consiste à effectuer l'opération suivante : $y = r + ts$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série, t étant un entier.

19. Procédé selon la revendication 18, dans lequel la relation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel la fonction de vérification compare le résultat fourni par la fonction appliquée à la valeur d'authentification V à une des valeurs suivantes : la valeur x, le produit de la valeur x avec la clé p publique de la puce correspondant à sa clé s secrète, en fonction de la valeur du paramètre t, ce qui revient à tester l'équation suivante : $g^y = xp^t$, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par la fonction $p = g^s$.

20. Procédé selon la revendication 1, dans lequel le paramètre x transmis de la puce à l'application est le résultat d'une fonction de hachage appliquée au moins à un élément relié à l'aléa r par une fonction mathématique et à un champ optionnel D contenant des données liées à l'application.

21. Procédé selon la revendication 20, dans lequel la fonction arithmétique a en outre pour arguments d'entrée des paramètres d'entrée et consiste à effectuer l'opération suivante : $y = r + 2^t s$ en fonction de la valeur attribuée par l'application à un

paramètre t d'entrée de la fonction série, ce paramètre t consistant en une chaîne de m bits ($t_{m-1},\ldots,t_0$) dont un seul des bits est égal à 1, le bit t_i, m étant un entier naturel.

22. Procédé selon la revendication 21, dans lequel la relation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel la fonction de vérification consiste à tester l'équation suivante : $h(g^y / p^{2^i}, D) = x$, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par la fonction $p = g^s$.

23. Procédé selon la revendication 21, dans lequel la relation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel la fonction de vérification consiste à tester l'équation suivante : $h(g^y.p^{2^i}, D) = x$, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par la fonction $p = g^{-s}$.

24. Procédé selon la revendication 20, dans lequel la fonction arithmétique a en outre pour arguments d'entrée des paramètres d'entrée et consiste à effectuer l'opération suivante : $y = r - 2^i s$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série, ce paramètre t consistant en une chaîne de m bits ($t_{m-1},\ldots,t_0$) dont un seul des bits est égal à 1, le bit t_i, m étant un entier naturel..

25. Procédé selon la revendication 24, dans lequel la relation mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel la fonction de vérification consiste à tester l'équation suivante : $h(g^y.p^{2^i}, D) = x$, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par la fonction $p = g^s$.

26. Procédé selon la revendication 20, dans lequel la fonction mathématique consiste en une fonction g^r dans un ensemble G d'éléments g muni d'une opération ayant au moins pour propriété d'être associative et dans lequel le paramètre x transmis de la puce à l'application est le résultat d'une relation du type

$x = h(g^r, D)$, où D désigne un champ optionnel contenant des données liées à l'application et h est la fonction de hachage.

27. Procédé selon la revendication 26, dans lequel la fonction série a pour arguments d'entrée des paramètres d'entrée, et consiste à effectuer une des opérations suivantes : $y = r$ et $y = r + s$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série et dans lequel la fonction de vérification compare la valeur x à une des valeurs suivantes : $h(g^y, D)$, $h(g^y / p, D)$, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par l'équation $p = g^s$.

28. Procédé selon la revendication 26, dans lequel la fonction série a pour arguments d'entrée des paramètres d'entrée, et consiste à effectuer une des opérations suivantes : $y = r$ et $y = r + s$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série et dans lequel la fonction de vérification compare la valeur x à une des valeurs suivantes : $h(g^y, D)$, $h(g^y.p, D)$, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par l'équation $p = g^{-s}$.

29. Procédé selon la revendication 26, dans lequel la fonction série a pour arguments d'entrée des paramètres d'entrée, et consiste à effectuer une des opérations suivantes : $y = r$ et $y = r - s$ en fonction de la valeur attribuée par l'application à un paramètre t d'entrée de la fonction série et dans lequel la fonction de vérification compare la valeur x à une des valeurs suivantes : $h(g^y, D)$, $h(g^y.p, D)$, en fonction de la valeur du paramètre t, y étant égale à la valeur d'authentification V et p étant la clé publique de la puce correspondant à sa clé secrète s, définie par l'équation $p = g^s$.

30. Procédé selon l'une des revendications 7 à 29, dans lequel l'ensemble G est le groupe Z_n^* des entiers positifs ou nuls inférieurs à n et premiers avec n.

31. Procédé selon l'une des revendications 7 à 29, dans lequel l'ensemble G est toute courbe elliptique construite sur tout corps fini quelconque.

32. Dispositif (6) à puce électronique permettant la mise en œuvre d'un procédé cryptographique asymétrique de protection contre la fraude de la puce

électronique selon l'une des revendications précédentes, dans des transactions entre une application et la puce électronique, consistant à calculer par la puce électronique une valeur V d'authentification à partir de paramètres d'entrée, ledit dispositif comprenant :

- un générateur (7) pseudo-aléatoire série produisant un aléa r propre à la transaction,

-un premier moyen (8) de mémorisation dans lequel est stocké au moins une valeur de x, la valeur du paramètre x étant calculée préalablement à la transaction par l'application et étant reliée à la valeur de l'aléa r par une relation mathématique,

-un moyen (9) de transmission de la puce électronique à l'application du paramètre x relié à l'aléa r propre à la transaction,

-un moyen (10) de mise en oeuvre d'une fonction série ayant pour paramètres d'entrées au moins l'aléa r propre à la transaction et une clé s privée appartenant à une paire de clés asymétrique (s, p) et fournissant en sortie un paramètre y,

-un moyen (9) de sortie pour constituer la valeur V d'authentification à partir d'au moins le paramètre y.

33. Dispositif de vérification pour la mise en œuvre d'un procédé cryptographique asymétrique de protection contre la fraude d'une puce électronique, dans des transactions entre une application et la puce électronique, selon l'une des revendications 1 à 31 consistant à vérifier une valeur V d'authentification calculée par la puce électronique à partir de paramètres exclusivement publics, ledit dispositif comprenant un moyen de mise en œuvre de la fonction de vérification prenant en entrée au moins la valeur V d'authentification et la clé p publique.

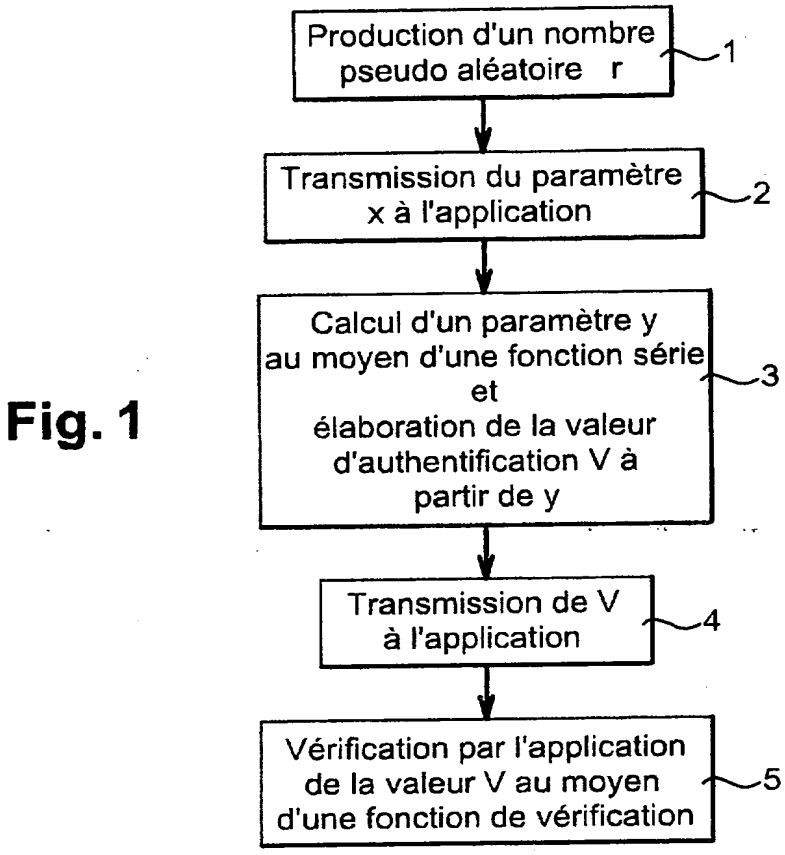
Production d'un nombre pseudo aléatoire r
1
Transmission du paramètre x à l'application
2
Calcul d'un paramètre y au moyen d'une fonction série et élaboration de la valeur d'authentification V à partir de y
3
Transmission de V à l'application
4
Vérification par l'application de la valeur V au moyen d'une fonction de vérification
5

Fig. 1

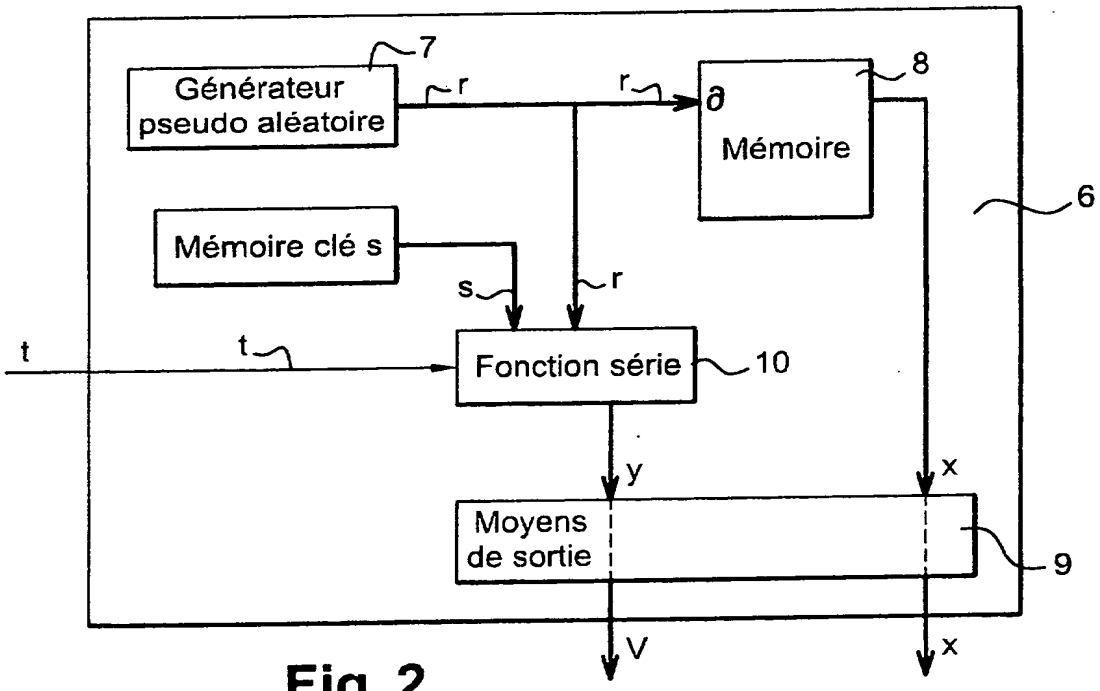
7
Générateur pseudo aléatoire
r
r
∂
Mémoire
8
6
Mémoire clé s
s
r
t
t
Fonction série
10
y
x
Moyens de sortie
9
V
x

Fig. 2

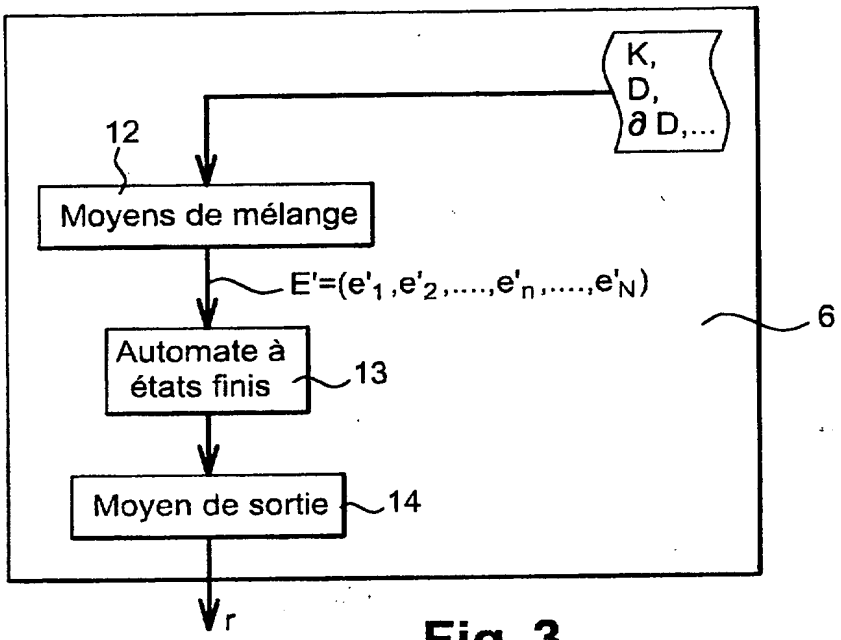
K,
D,
∂ D,...
12
Moyens de mélange
E'=(e'1,e'2,....,e'n,....,e'N)
6
Automate à états finis
13
Moyen de sortie
14
r

Fig. 3

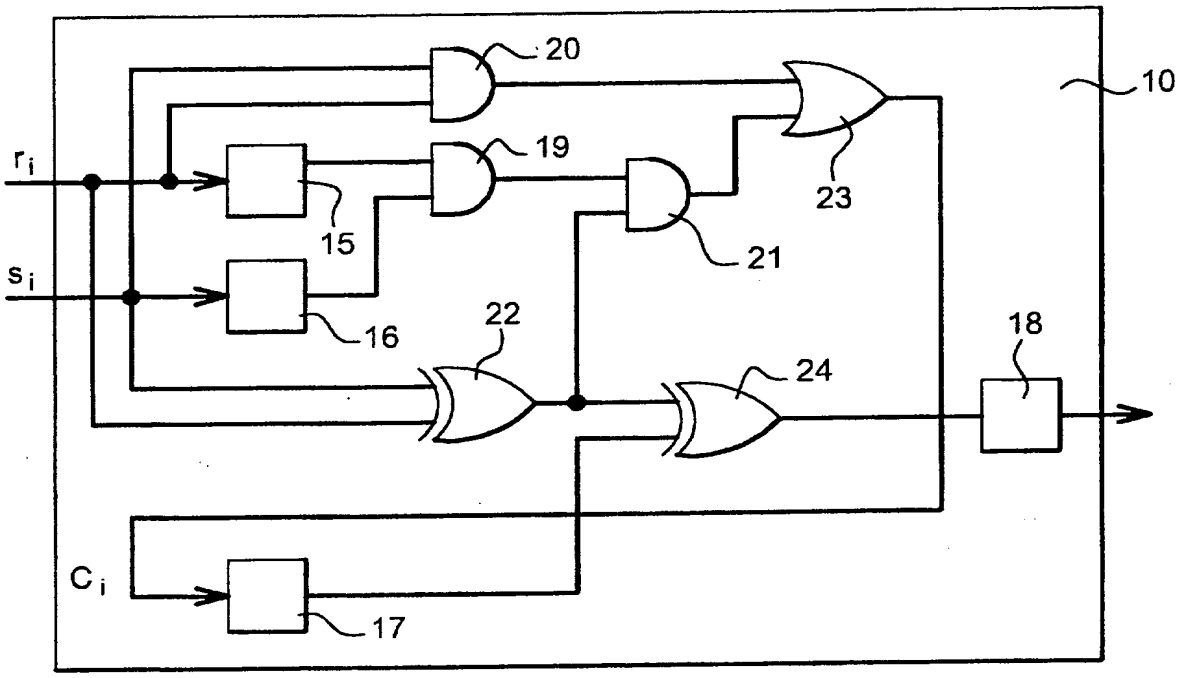
20
10
ri
19
23
15
21
si
22
18
16
24
Ci
17

Fig. 4

INPI
INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

DÉPARTEMENT DES BREVETS

26 bis, rue de Saint Pétersbourg
75800 Paris Cedex 08
Téléphone : 01 53 04 53 04 Télécopie : 01 42 93 59 30

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

Code de la propriété intellectuelle - Livre VI

cerfa N° 11 235*02

DÉSIGNATION D'INVENTEUR(S) Page N° 1 / 1

(Si le demandeur n'est pas l'inventeur ou l'unique inventeur)

Cet imprimé est à remplir lisiblement à l'encre noire — DB 113 W /260899

Vos références pour ce dossier *(facultatif)*	04539
N° D'ENREGISTREMENT NATIONAL	0 30 1108

TITRE DE L'INVENTION (200 caractères ou espaces maximum)

Procédé cryptographique à clé publique pour la protection d'une puce contre la fraude

LE(S) DEMANDEUR(S) :

FRANCE TELECOM
Société Anonyme
6, place d'Alleray
75015 PARIS

DESIGNE(NT) EN TANT QU'INVENTEUR(S) : (Indiquez en haut à droite «Page N° 1/1» S'il y a plus de trois inventeurs, utilisez un formulaire identique et numérotez chaque page en indiquant le nombre total de pages).

Nom		GIRAULT
Prénoms		Marc
Adresse	Rue	4, rue Vivianc
	Code postal et ville	14000 CAEN
Société d'appartenance *(facultatif)*		FRANCE TELCOM
Nom		
Prénoms		
Adresse	Rue	
	Code postal et ville	
Société d'appartenance *(facultatif)*		
Nom		
Prénoms		
Adresse	Rue	
	Code postal et ville	
Société d'appartenance *(facultatif)*		

DATE ET SIGNATURE(S) DU (DES) DEMANDEUR(S) OU DU MANDATAIRE (Nom et qualité du signataire)

JEUNE Pascale
Mandataire par pouvoir PG 8611

24/01/03